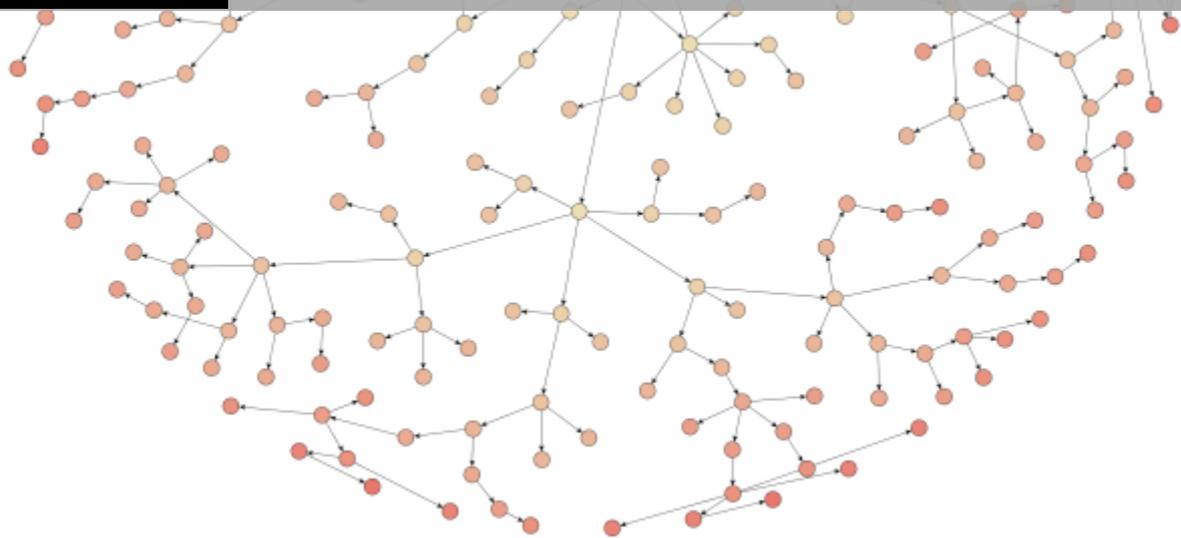


WHITE PAPER

MODEL BASED TESTING



Finance Institution ROI for implementing MBT automation

## Case: Finance Institution's return of investment for implementing Model-Based test automation

### Executive summary

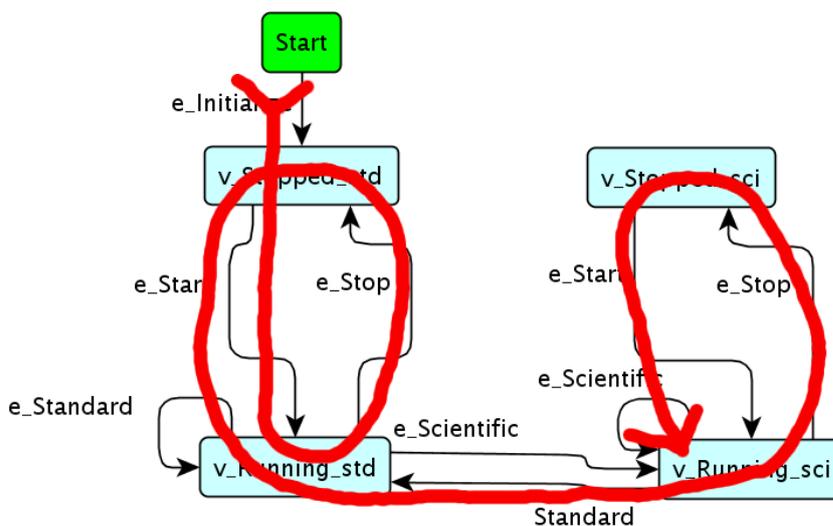
The Company implemented test automation with MBT (Model-based testing) and saved est. 2000h in decreased error reporting alone and gained several other benefits including better collaboration, better productivity and better test coverage.

### Background

The Company implemented MBT in the spring of 2008 to be able to keep up with the new agile software development method (Scrum). Quickly realizing that classic test automation never could keep the pace of 4 parallel scrum team. The Company searched for a faster and more aligned way of automation.

### Looking into MBT

MBT uses a model to describe the system instead of traditional test cases and the execution of a "test" equals traversing the model (there are several techniques for traversing models depending of situation, targets, type of application etc.). A simple graph is shown below illustrating one way of running through the graph with the goal to cover all vertexes (the blue colour boxes, indicating application states).



Powered by yFiles

There are several areas where automation with MBT differs from a more traditional approach:

- The Graph. The graph increases the total understanding of the application under test and exposes an opportunity to identify flaws/errors and misinterpretations even *before* any actual testing have started. It also makes it easier to discuss any found errors/bugs since the graph is a natural visual aid to clarify how a bug was discovered (instead of looking into source code etc).

- Test coverage. Instead of running vanilla regression tests in silo-like thinking MBT test the application as it is described in the graph. Most likely you will run your MBT test for total coverage and possibly for a prolonged time. By doing that you will cover all possible user scenarios jumping back and forth with test coverage previously unreachable, all with the same automation effort!
- Time to first test. As soon as the first steps in the graph and the according test automation scripting are implemented you can start to test! This significantly increases the usage, usefulness and effectiveness of test automation.
- Keeping up with Agile tact time. Traditional test automation struggles to keep up with the changes in functionality and scope in modern agile development where MBT in many cases only require re-drawing in the graph and possibly some additional re-coding.
- Maintenance. Since the test automation code is divided into small pieces (boxes and arrows in the graph) the maintenance is a simple and straight forward approach. It is easy to find *where* we need to change something and while updating some parts we can still execute test in the rest of the application/graph.
- Continuous test gives continuous feedback. As soon as some parts of the MBT is implemented you start to test and probably during continuous/nightly build. This gives fast feedback to developers and shortcut most of the traditional error reporting documents, meetings and discussions.

### *ROI for MBT*

Looking at the ROI for The Company's implementation of MBT they only looked at the last benefit and found that they saved 2000h during an 8 month period just because they didn't need the administrative work of error/bug handling. Of course some found defects were so severe that they indeed needed documentation and meetings etc. but they were reduced to 68 where the total amount for such a big project should be roughly 1100 for a 4 scrum team project (based on statistics from previous projects).

Other non-quantified but recognized benefits:

- Non-technical testers could automate by creating the graph(s) and technical testers could focus on implementing automation code. This also had the side-effect that non-technical testers enjoyed their work more simply because drawing graphs is more fun than writing test cases.

- Having discussions about functionality, bugs etc. were much more effective when the graph could be used as a visual aid, bridging the gap between developers and non-technical team members as well as end-users.

### *Things to look out for*

Remember that the graphs used in MBT should belong to test, or at least to the subject matter experts. Do not use the developers models (if they have any) since you will in that case just test what's been developed, not necessary what the organization wanted.

**P**rolore has quality assurance in focus and testing tools as a speciality. We employ the best and brightest consultants in the testing and quality assurance field with the highest level of seniority and a wide range of industry experience. So when you require our expert resources to carry out testing services on your mission-critical projects, you can be assured that we will deploy the highest-calibre people who will 'hit the ground running' and add value quickly. Key service areas are mentoring, training, outsourcing, organization and processes within test management, test automation, performance test, monitoring and metrics.